EXERCISE:

Task 01

Write MATLAB Code for DSB Modulation

$$yDSB = Am\cos(2\ fmt)\ Vc\cos(2\ fct)$$

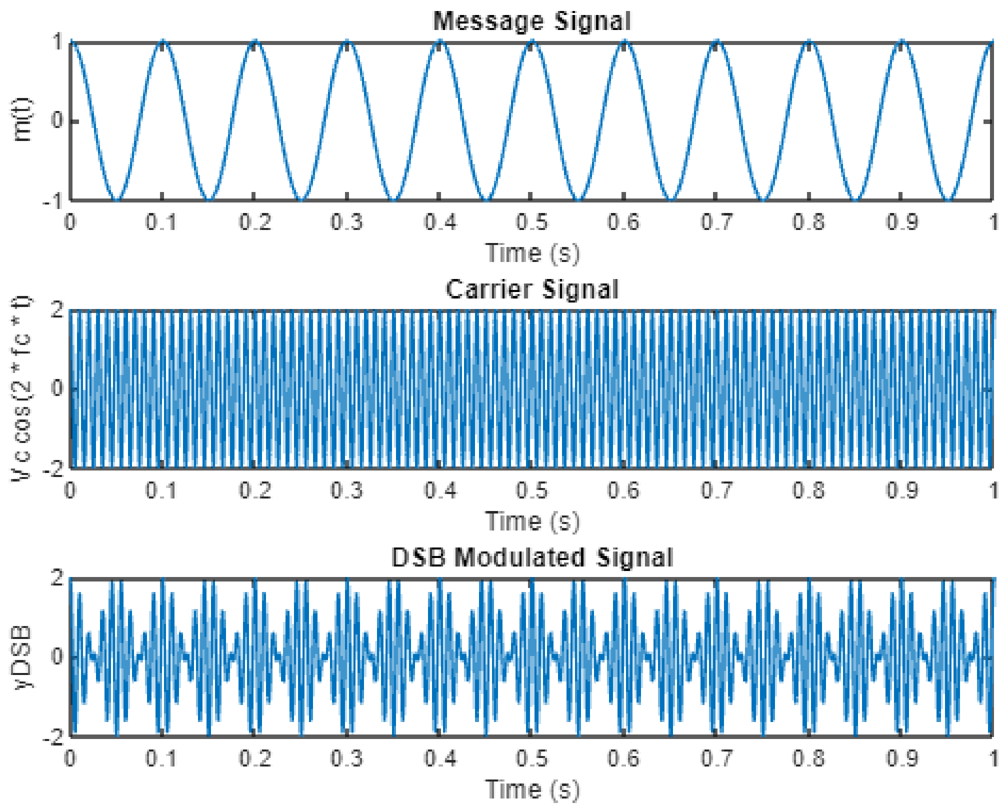
Where $Am = 1$ and $fm = 100Hz$, $Vc = 2$ and $fc = 10kHz$

- Plot $m(t)$, Vc cos (2 fc t) and yDSB in time domain for t = 0 : 1/fs:1, where $fs = 1000Hz$
- Plot frequency spectrum of m(t), $Vccos(2\ fct)$ and $yDSB$ in frequency domain and reconstruct demodulated signal using filter.
- What can you analyze from the frequency components? Are they consistent with your theoretical knowledge?

```
% Parameters
Am = 1;
fm = 10; % in Hz
Vc = 2;
fc = 100; % in Hz
fs = 1000; % in Hz
t = 0:1/fs:1;

% Signals
mt = Am * cos(2 * pi * fm * t);
Vc_cos = Vc * cos(2 * pi * fc * t);
yDSB = Am * cos(2 * pi * fm * t) .* Vc_cos;

% Plot time domain signals
figure;
subplot(3,1,1);
plot(t, mt);
xlabel('Time (s)');
ylabel('m(t)');
title('Message Signal');
subplot(3,1,2);
plot(t, Vc_cos);
xlabel('Time (s)');
ylabel('Vc cos(2 * fc * t)');
%xlim([0 0.2])
title('Carrier Signal');
subplot(3,1,3);
plot(t, yDSB);
xlabel('Time (s)');
ylabel('yDSB');
title('DSB Modulated Signal');
```

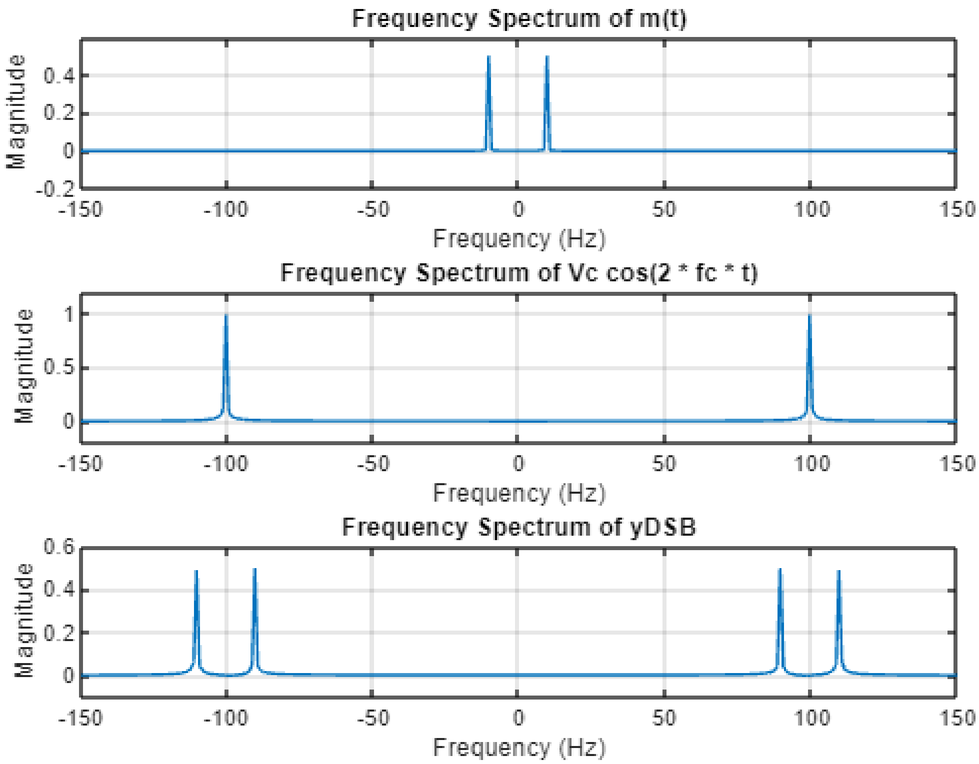



```
%xlim([0 0.5])

% Plot frequency domain signals
f = -fs/2:fs/length(t):fs/2-fs/length(t);
MT = fftshift(fft(mt));
VC_COS = fftshift(fft(Vc_cos));
YDSB = fftshift(fft(yDSB));
figure;
subplot(3,1,1);
plot(f, abs(MT)/length(t));
xlabel('Frequency (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum of m(t)');
xlim([-150 150])
grid("on")
ylim([-0.2 0.6])

subplot(3,1,2);
plot(f, abs(VC_COS)/length(t));
xlabel('Frequency (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum of Vc cos(2 * fc * t)');
xlim([-150 150])
ylim([-0.2 1.2])
grid("on")
```

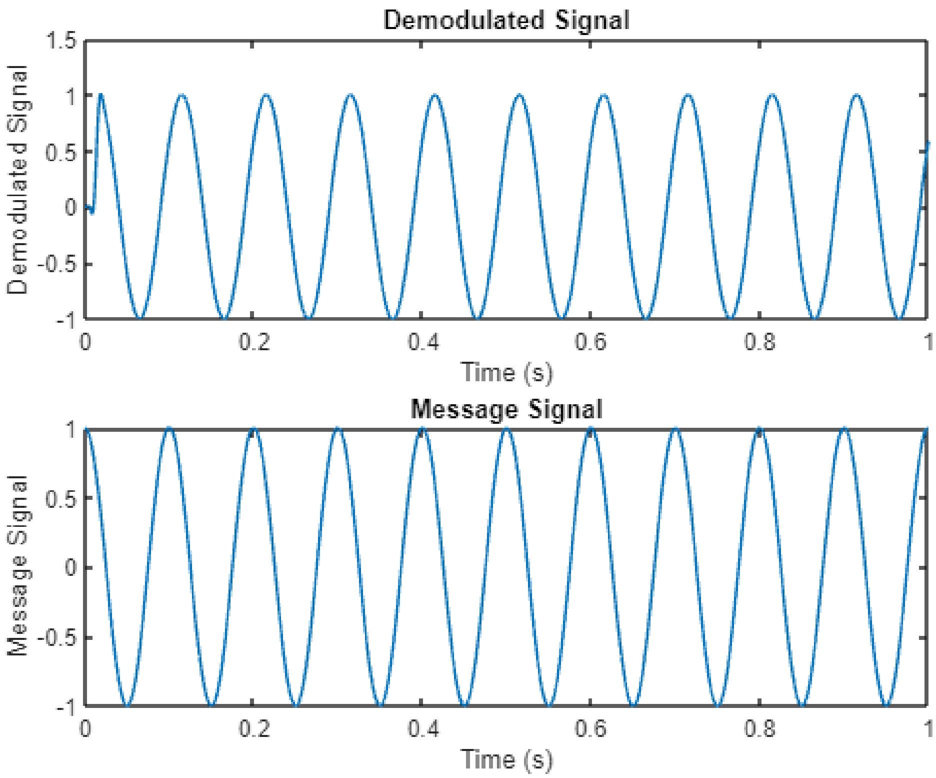
```
subplot(3,1,3);
plot(f, abs(YDSB)/length(t));
xlabel('Frequency (Hz)');
ylabel('Magnitude');
title('Frequency Spectrum of yDSB');
xlim([-150 150])
ylim([-0.1 0.6])
grid("on")
```




```
% Reconstruct demodulated signal using filter
y1=yDSB.*Vc_cos;
h = fir1(30,   fc/(fs/2)); % Design a lowpass filter
y_demod =0.5* filter(h, 1, y1); % Demodulate using the filter

% Plot demodulated signal
figure;
subplot(2,1,1)
plot(t, y_demod);
xlabel('Time (s)');
ylabel('Demodulated Signal');
title('Demodulated Signal');
subplot(2,1,2)
plot(t, mt);
xlabel('Time (s)');
```

```
ylabel('Message Signal');
title('Message Signal');
```



Explanation:

I set up some parameters for a communication system. I have a message signal, a carrier signal, and a modulated signal. Then, I plotted these signals to see how they look over time and in terms of their frequency. After that, I demodulated the modulated signal using a filter to recover the original message signal. Finally, I compared the demodulated signal with the original message signal by plotting them together. This code helps me understand how communication signals are processed and how modulation and demodulation work in practical scenarios.

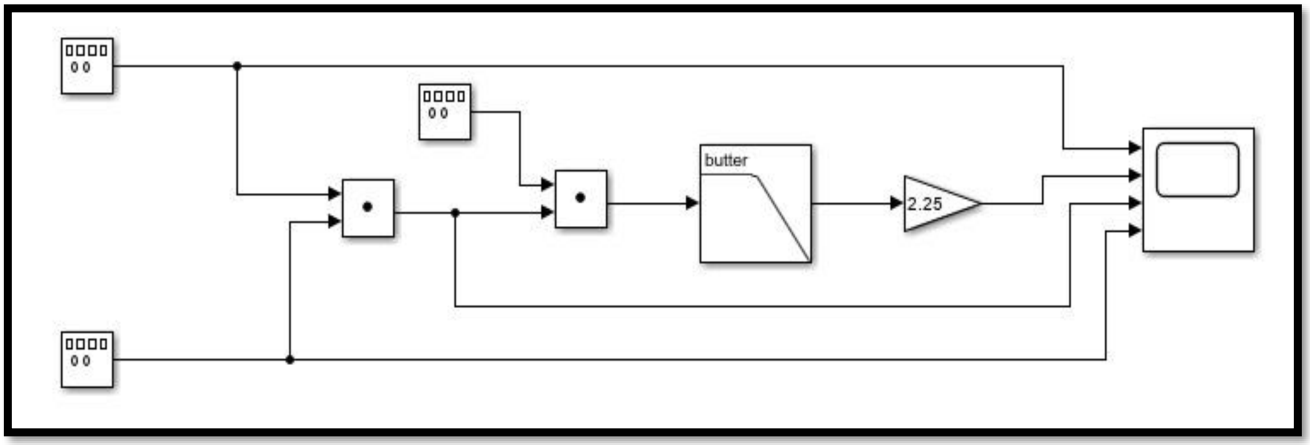
Task 02

Implement DSB-SC Modulation and De-modulation using Simulink, you can use the following blocks in Simulink to implement it.

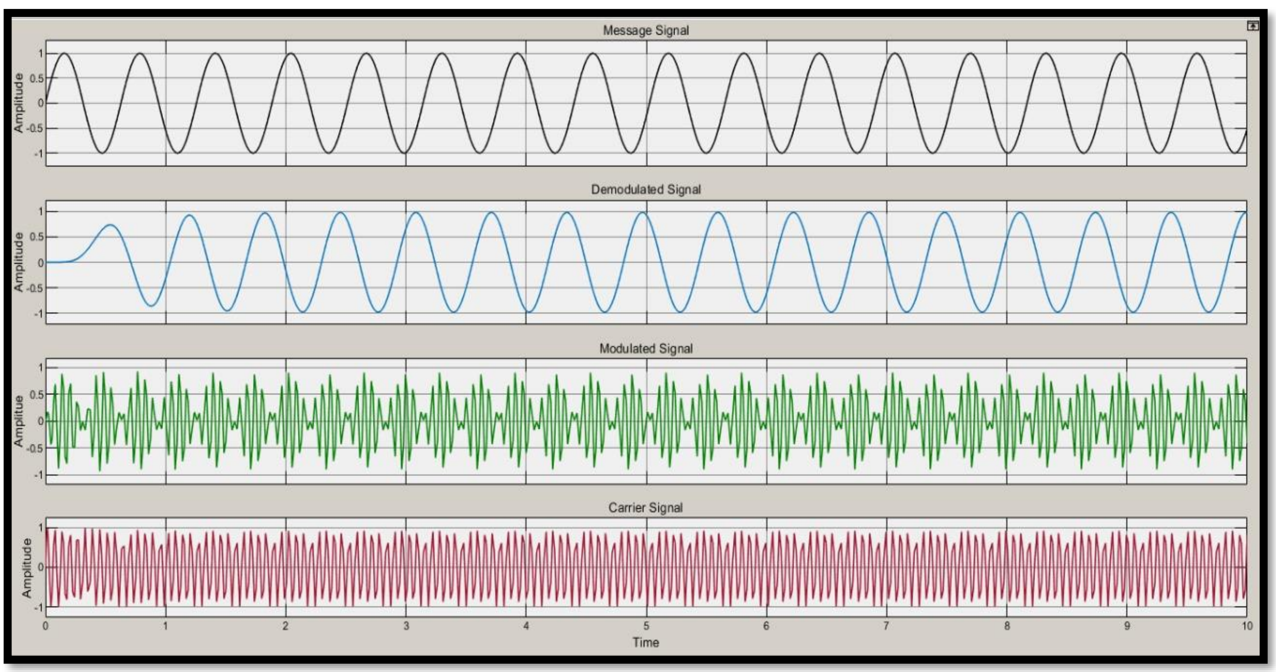
Where fm = 10Hz and fc = 100Hz

- Signal Generator (Carrier and Message Generation)
- Dot Product (for multiplication)
- Analog Filter Design (Butter-low pass)
- Gain (use at Demodulation side)
- Scope

Block Diagram:



Output:



Explanation:

To implement DSB-SC (Double Sideband Suppressed Carrier) Modulation and De-modulation using Simulink, I used several blocks: Signal Generator for generating both carrier and message signals with frequencies of 10 Hz and 100 Hz respectively. Then, I used the Dot Product block for multiplying the message signal with the carrier signal. Next, I utilized the Analog Filter Design block, specifically a Butterworth low-pass filter, to filter the modulated signal. At the demodulation side, I used the Gain block to adjust the amplitude of the demodulated signal. Finally, I used the Scope block to visualize the signals at various stages of modulation and demodulation. This setup in Simulink demonstrated the process of DSB-SC modulation and demodulation.

Conclusion:

In conclusion, I successfully implemented DSB-SC (Double Sideband Suppressed Carrier) Modulation and De-modulation using Simulink. I utilized various blocks within Simulink, including Signal Generator, Dot Product, Analog Filter Design, Gain, and Scope, to carry out the modulation and demodulation processes. By generating carrier and message signals, multiplying them, filtering the modulated signal, and adjusting the amplitude of the demodulated signal, I effectively demonstrated the principles of DSB-SC modulation and demodulation. This exercise enhanced my understanding of communication systems and signal processing techniques, providing valuable insights into practical applications of modulation and demodulation in engineering and telecommunications.